RX23E-A Group

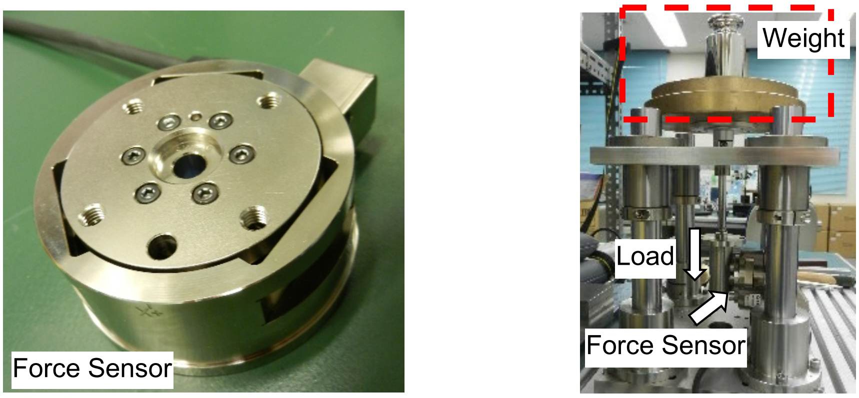
Force Sensor Measurement Example

Overview

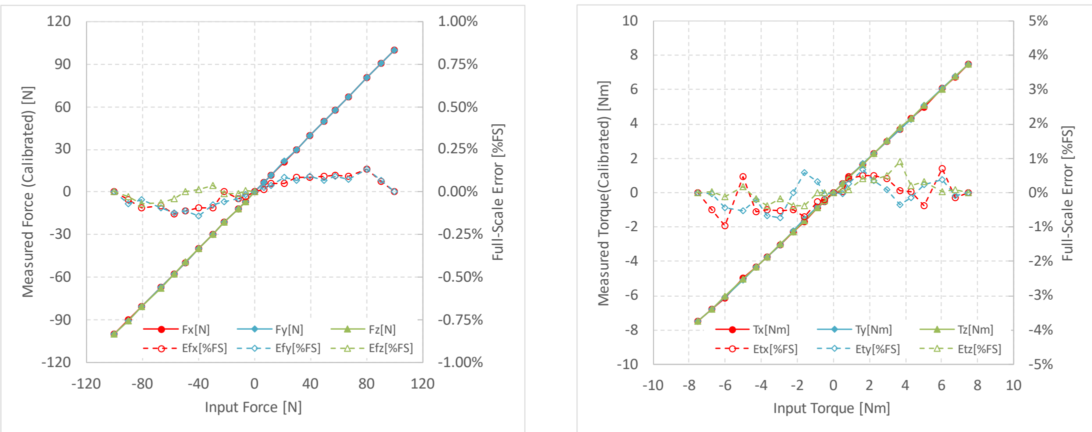
This document describes the example of the program which obtains three-dimensional force and torque by the strain gauge based 6-axis force sensor, using RX23E-A. This example uses two units of DSAD to obtain output from six channels of the force sensor by scanning three channels with one unit of DSAD. We have measured the force sensor with this program. The appearances of the force sensor and the evaluation environment, and the evaluation results are shown below.

Target Device

RX23E-A



Appearance of Force Sensor and Evaluation Environment



Result of Force Measurement (Left) and Torque Measurement (Right)

Measurement Uncertainty

Item	$E_{Fx:FS}$ [%FS]	$E_{Fy:FS}$ [%FS]	$E_{Fz:FS}$ [%FS]	$E_{Tx:FS}$ [%FS]	$E_{Ty:FS}$ [%FS]	$E_{Tz:FS}$ [%FS]
9105-TWE-Gamma SI-130-10 Measurement uncertainty (95% CI)	1.00%	1.25%	0.75%	1.00%	1.25%	1.50%
Result of full-scale error measurement (Worst case)	0.13%	0.14%	0.07%	0.95%	0.68%	0.89%

Contents

1. Overview

This document describes the example of the program to obtain three-dimensional force and torque using the strain gauge based 6-axis force sensor by RX23E-A. This example uses two units of DSAD to obtain the output from six channels of the force sensor by scanning three channels with one unit of DSAD. The sample program runs on the Renesas Solution Starter Kit for RX23E-A (RSSKRX23E-A) board. The measurement results can be displayed on the PC tool program V2.0 of RSSKRX23E-A.

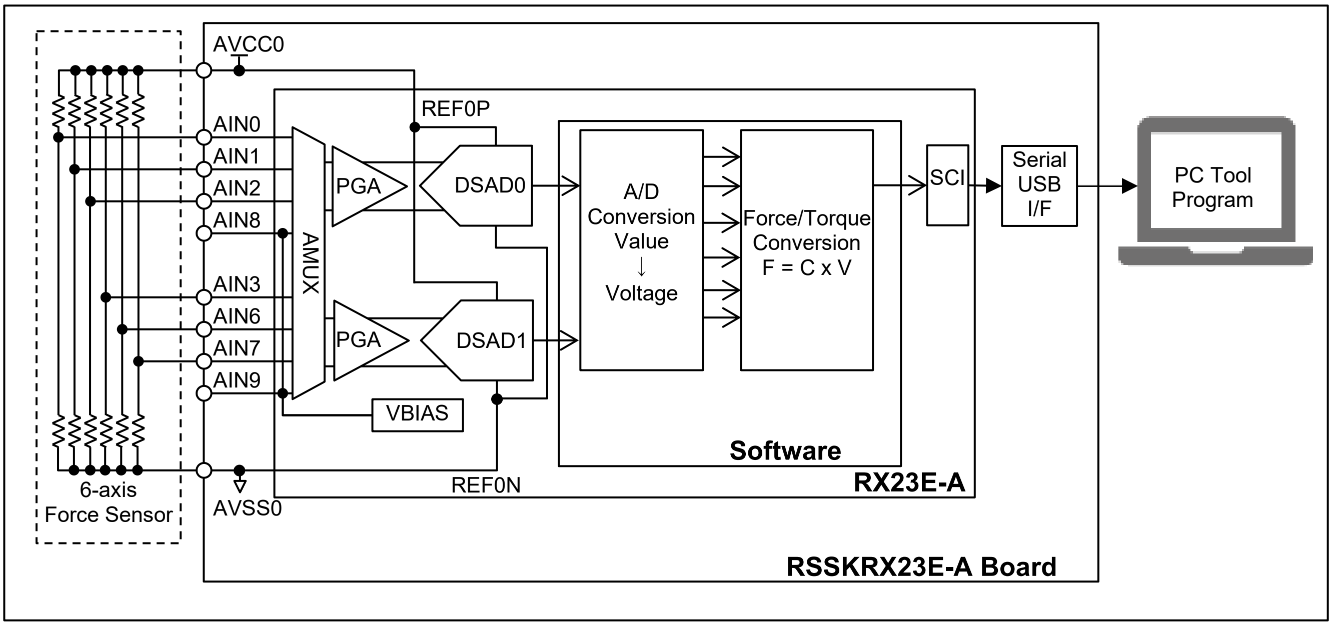


Figure 1-1 6-axis Force Sensor Measurement System with RSSKRX23E-A Board

2. Related Documents

- R01UH0801 RX23E-A Group User’s Manual: Hardware
- R20UT4542 RSSKRX23E-A User’s Manual
- R20AN0540 Application Notes RSSKRX23E-A PC Tool Program Operation Manual
- R01AN4799 Application Notes RX23E-A Group Effective Use of AFE and DSAD
- R01AN4359 RX family RX DSP Library Version 5.0

3. Environment for Operation Confirmation

Table 3-1 shows the environment to check the operation.

Table 3-1 Environment for Operation Check

Item	**Description**	
Board	RSSKRX23E-A Board（RTK0ESXB10C00001BJ）	
MCU	RX23E-A（R5F523E6ADFL） Power-supply voltage (VCC, AVCC0)：5V Operating frequency (ICLK)：32MHz Peripheral operating frequency (PCLKB)：32MHz DSAD operating frequency (f_{DR})：4MHz DSAD modulator clock frequency (f_{MOD})：0.5MHz	
Force sensor	Manufacturer	ATI Industrial Automation
	Model	9105-TWE-Gamma
	Calibration	SI-130-10
	Measurement uncertainty [%FS] (95% CI)	Fx: 1.00%, Fy: 1.25%, Fz: 0.75% Tx: 1.00%, Ty: 1.00%, Tz: 1.50%
IDE	Renesas e^2 Studio Version 2021-10 Renesas RX Smart Configurator V2.11.0	
Tool Chain	Renesas CC-RX V3.03.00	
Emulator	E2 Emulator Lite	

4. Force Sensor Measurement

Figure 4-1 shows the connection of the force sensor and the RSSKRX23E-A board. Red letters in the figure indicate parts to be changed. The details of the changed parts are shown in Table 4-1

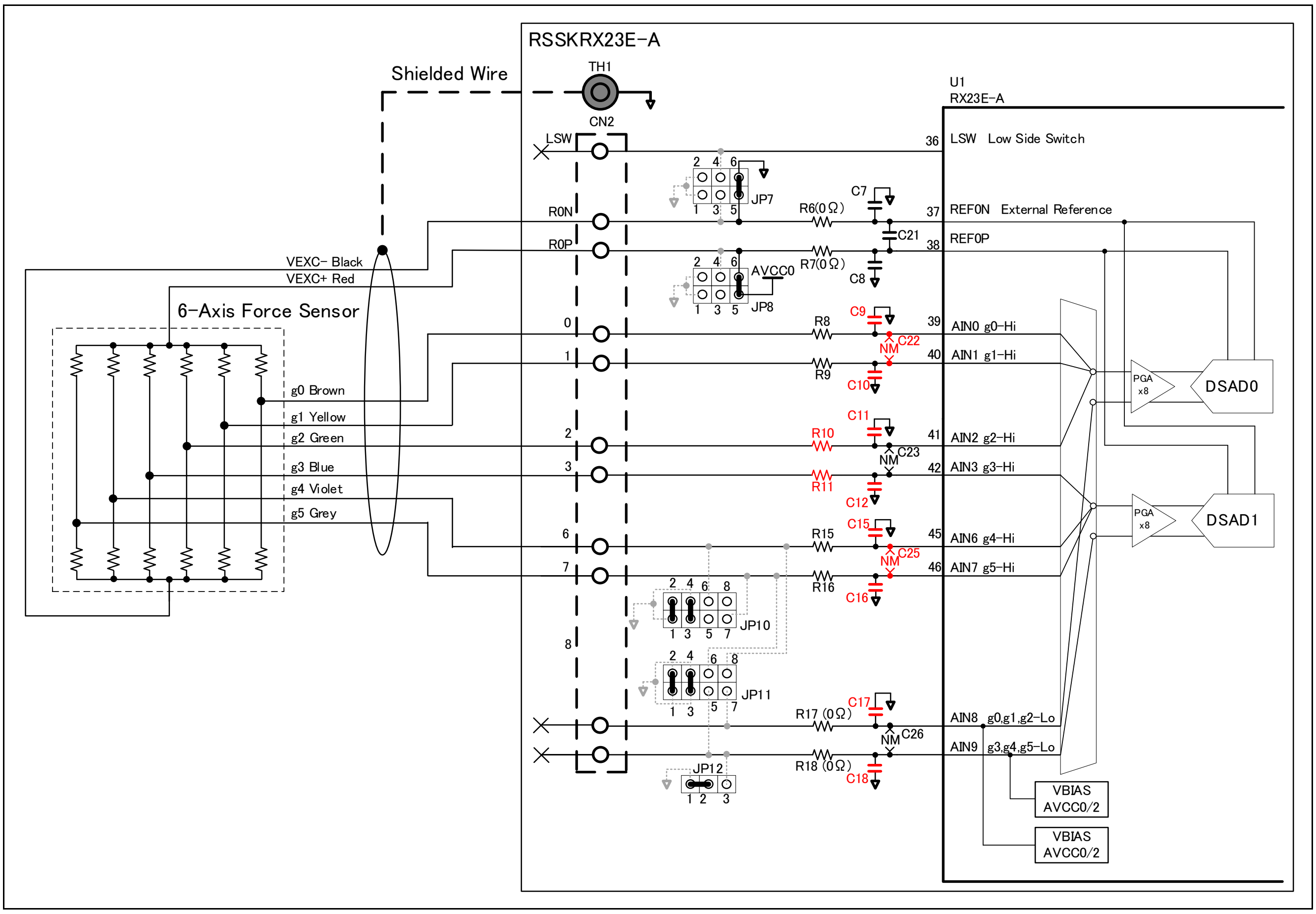


Figure 4-1 Connection of RSSKRX23E-A Board and Force Sensor

Table 4-1 Area of Changed Parts of RSSKRX23E-A

Circuit Designator	**Value (Before the change)**	**Value (After the change)**
C22, C25	0.1µF 50V	Not mounted
C9, C10, C15, C16	0.01µF 50V	1µF 25V
C11, C12, C17, C18	Not mounted	1µF 25V
R10, R11	0Ω 1A	1kΩ ±1%

When a voltage is applied to the excitation voltage terminal of a force sensor, the force sensor outputs the potential of midpoint of the half-bridge resistors which are connected to strain gauge in series. The output of the force sensor is connected to AIN0, AIN1, AIN2, AIN3, AIN6, and AIN7 of RX23E-A. The channel function of DSAD on RX23E-A is used for measurement, and DSAD0 is used for voltage measurement on AIN0, AIN1, and AIN2, and DSAD1 is used for voltage measurement on AIN3, AIN6, and AIN7. AIN8 is used for input for DSAD0 Lo side, and AIN9 is used for input for DSAD1 Lo side. By outputting VBIAS to AIN8 and AIN9 for each, the voltages on AIN8 and AIN9 are set to a half voltage of AVCC0, which is equivalent to the output voltage of the force sensor at no load.

4.1 Force Sensor

The strain gauge type 6-axis force sensor is a sensor that utilizes the fact that the resistance value of each strain gauges mounted on the strain body changes due to stress. By applying a voltage to the 6-axis force sensor, the change in resistance value due to stress is measured as a voltage.

If the output voltage of the strain gauge is non-linear in relation to the stress, the characteristic curve is divided into multiple regions and linear approximation, for example, is performed in each of the regions to increase the measurement precision, thereby matching the characteristic curve. In this example, the region is regarded as a single linear characteristic without being divided, and the voltage is converted to strain amount with linear interpolation.

Supposing that the applied voltage to the strain gauge is V_{CC}, the rated output is RO, and the load rating is S_{max}, the output voltage V of applied strain S is calculated as below.

$$V = \mathrm{RO} \cdot V_{cc} \cdot \frac{S}{S_{max}}$$

Multiply the acquired 6-axis voltage to the force sensor-specified voltage-load conversion matrix C to calculate the force and torque on x, y and z axis.

$$F = C \times V$$

$$\begin{pmatrix} F_x \\ F_y \\ F_z \\ T_x \\ T_y \\ T_z \end{pmatrix} = \begin{pmatrix} C_{11} & \cdots & C_{16} \\ \vdots & \ddots & \vdots \\ C_{61} & \cdots & C_{66} \end{pmatrix} \begin{pmatrix} V_0 \\ V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \end{pmatrix}$$

In this example, ATI Industrial Automation 9105-TWE-Gamma is used as a force sensor for measurement. The appearance of the force sensor is shown in Figure 4-2.

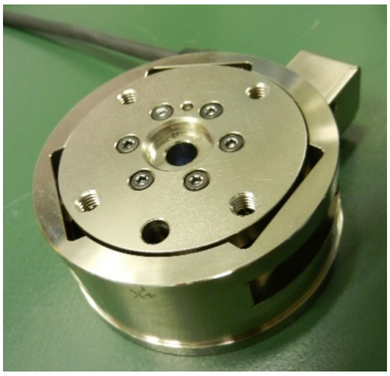


Figure 4-2 Appearance of ATI Industrial Automation 9105-TWE-Gamma

4.2 A/D Conversion of Strain Gauge Output

This example uses the supplied voltage of the strain gauges for reference voltage as shown in Figure 4-1. The output voltage of each strain gauge is A/D converted by DSAD.

Table 4-2 shows the measurement conditions of the strain gauges. The digital filter of the DSAD generates gain of from 1/2 to 1 time when oversampling ratio is not a power of two. A/D conversion value is treated as affected by this gain.

Table 4-2 Measurement Condition of Strain Gauge

Item	Condition	Remarks
PGA gain G_{PGA}	x8	
DSAD reference voltage V_{REF}	5V	Applied voltage of the strain gauge (REF0P=AVCC0, REF0N=AVSS0)
Oversampling ratio OSR	32	
Digital filter gain G_{DF}	1.0	$G_{DF} = 1/2^{(Ceil(4\log_2 OSR) - 4\log_2 OSR)}$
DSAD output format	2's Complement	

This example uses two units of DSAD on RX23E-A to scan the output from the 6-axis force sensor 3 voltages in each DSAD. Figure 4-3 shows conversion sequence and Table 4-3 shows A/D conversion time.

When starting the A/D conversion, use the synchronous start function to start the conversion of DSAD0 and DSAD1 simultaneously.

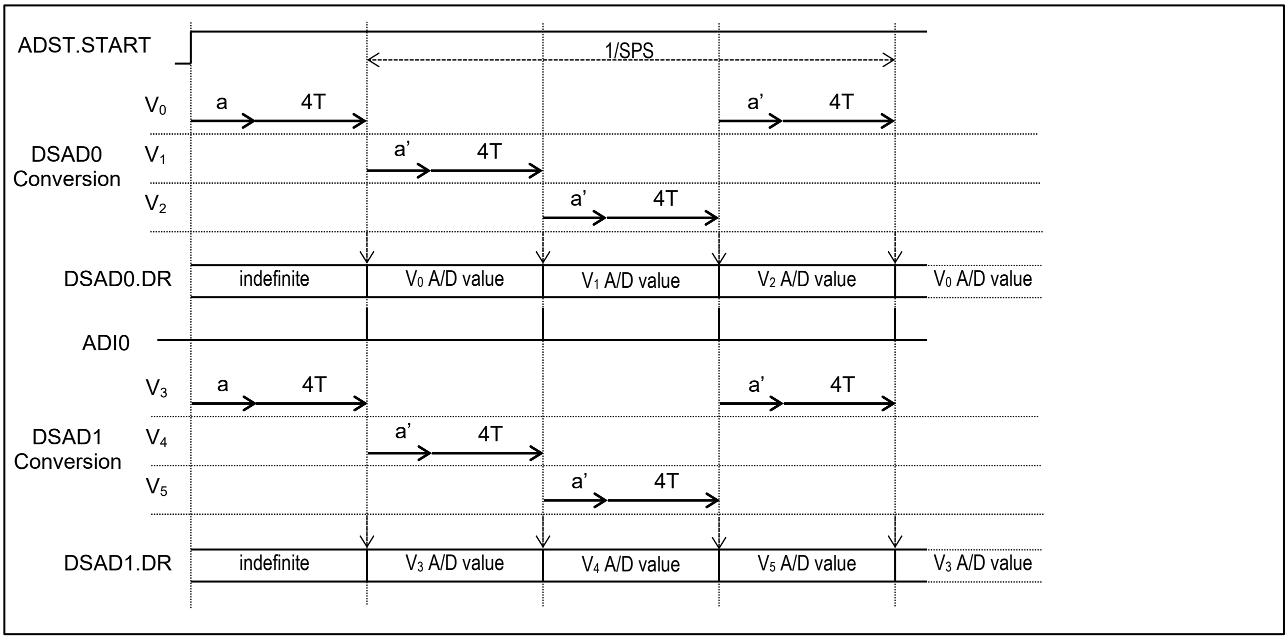


Figure 4-3 A/D Conversion Sequence and A/D Conversion Time

Table 4-3 A/D Conversion Time

Normal Mode: f_{MOD} = 0.5MHz
Over Sampling Ratio (OSR) = 32

Item		Value	Remarks
A/D Conversion Time		0.512[msec]	a' + 4T
	a	0.259[msec]	Average time of channel switching and stabilization
	a'	0.256[msec]	
	T	0.064[msec]	Digital filter processing time T = OSR / f_{MOD}
Data rate		651.0416667[sps]	1 / total A/D conversion time = 1 / 3(a'+4T)

4.3 Calculation Procedure

Follow the procedure below to convert from the A/D conversion value to the force and torque.

(1) Calculate the Voltages

Convert the A/D conversion value of the voltage outputted from the individual strain gauge into voltages.

Supposing that the PGA gain is G_{PGA}, the digital filter gain is G_{DF}, the reference voltage of the DSAD is V_{REF}, and the A/D conversion value is $DATA_n$, output voltage V_n from each strain gauge is calculated from the DSAD resolution of 24bit by the equation below.

$$\begin{aligned} V_n &= \frac{2V_{REF}}{2^{24} \cdot G_{PGA} \cdot G_{DF}} \cdot \mathrm{DATA}_n \\ &= \frac{V_{REF}}{2^{23} \cdot G_{PGA} \cdot G_{DF}} \cdot \mathrm{DATA}_n, \qquad V_{REF} = AVCC0 - AVSS0, \qquad n = 0 \sim 5 \end{aligned}$$

(2) Conversion of Force and Torque

Multiply the acquired 6-axis voltage to the force sensor-specified voltage-load conversion matrix C to calculate the force and torque on x, y and z axis.

$$F = C \times V$$

$$\begin{pmatrix} F_x \\ F_y \\ F_z \\ T_x \\ T_y \\ T_z \end{pmatrix} = \begin{pmatrix} C_{11} & \cdots & C_{16} \\ \vdots & \ddots & \vdots \\ C_{61} & \cdots & C_{66} \end{pmatrix} \begin{pmatrix} V_0 \\ V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \end{pmatrix}$$

4.4 Zero-Reset

To correct mechanical offset etc., the A/D conversion value at no load is adjusted to be zero.

In this example, supposing that the offset value is the average of A/D conversion values of individual strain gauge at no load, set the offset value in DSAD offset correction register OFCRm so that the offset is canceled.

5. Sample Program

5.1 Operation Overview

Figure 5-1 shows the processing flow of the sample program.

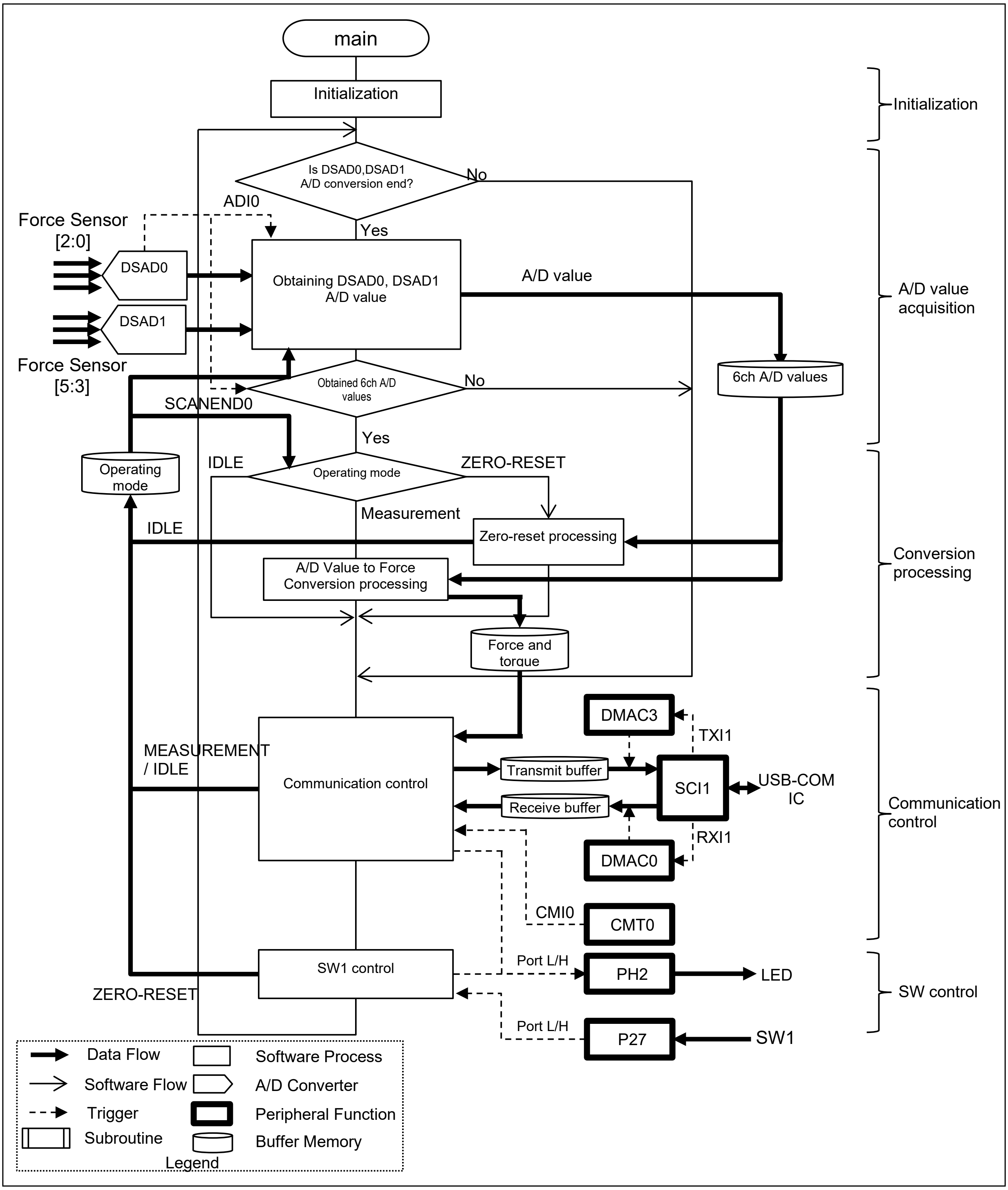


Figure 5-1 Force Sensor Measurement Process Flow

This sample program operates in the three operating modes: IDLE, MEAUSUREMENT, and ZERO-RESET. The operating mode is switched by a RSSKRX23E-A PC tool program and SW1 on RSSKRX23E-A board. Table 5-1 shows the transition of the operating mode.

Table 5-1 Operating Mode Transition

Operating mode	Operation	Transition trigger	Transition to
IDLE	No operation	Receive Run command	MEASUREMENT
		Press SW1	ZERO-RESET
MEASUREMENT	Force sensor measurement	Receive STOP command	IDLE
ZERO-RESET	Zero-reset processing	Complete Zero-reset processing	IDLE

The outline of each processing is described below.

- Initial setting
  The following initial settings are performed at startup.
  - Initial setting of DMAC to be used for communication (if a connection is made to the PC tool program of RSSKRX23E-A)
  - Initialization of the communication buffer and start of SCI1 operation (if a connection is made to the PC tool program of RSSKRX23E-A)
- A/D conversion value acquisition
  With the completion of the A/D conversion of both DSAD0 and DSAD1 as a trigger, acquires the A/D conversion values.
- When the A/D conversion values of 6ch are acquired, performs processing in the operating mode below.
  - MEASUREMENT: Based on "4.3 Calculation Procedure", calculates the force and torque from the A/D conversion values of 6ch.
  - ZERO-RESET: After setting each average of the A/D conversion values of 6ch in the register OFCRm on the corresponding channel, changes its operating mode to IDLE.
    For details, refer to "5.3 Zero-Reset Processing".
- Communication control
  For communication with the PC tool program of RSSKRX23E-A, the followings are processed. For details, refer to "5.4 Communication Control".
  - If a receive packet exists, analyzes it, performs processing corresponding to a command, and stores a reply packet in the transmit buffer.
  - If the measurement results are updated, stores a transmit packet in the transmit buffer.
  - If the transmit buffer contains un-transmitted data, starts transmission.
- SW1 control
  At the detection of pressing of SW1, the followings are processed if the operating mode is IDLE. For details, refer to "5.3 Zero-Reset Processing".
  - Sets DSAD for Zero-reset processing and starts A/D conversion.
  - Changes the operating mode to ZERO-RESET.

5.2 Peripheral Functions and Pins to be Used

The peripheral functions to be used in this example are listed in Table 5-2, the pins to be used in Table 5-3. Also, setting conditions for the peripheral functions are shown together.

The setting for the peripheral functions is used the Smart Configurator (hereinafter, called “SC”).

Table 5-2 Peripheral Functions to be Used

Peripheral Function	Purpose
AFE、DSAD0、DSAD1	Measurement of force sensor
SCI1	UART communication with PC tool program
DMAC0	Transmit data with reception completion interrupt of SCI1 as a trigger.
DMAC3	Transmit data with buffer empty interrupt of SCI1 as a trigger
CMT0	Detect communication time-out of SCI1
PH2	Control lighting of LED1
P27	Reading stare of SW1

Table 5-3 Pins to be Used

Pin name	I/O	Purpose
PH2	Output	LED1 lighting control
P27	Input	SW1 input
P26/TXD1	Output	UART1 transmission pin
P30/RXD1	Input	UART1 reception pin
P31/CTS1#	Input	CTS signal input pin
AIN0	Input	Input pin for sensor output on 0 Hi side
AIN1	Input	Input pin for sensor output on 1 Hi side
AIN2	Input	Input pin for sensor output on 2 Hi side
AIN3	Input	Input pin for sensor output on 3 Hi side
AIN6	Input	Input pin for sensor output on 4 Hi side
AIN7	Input	Input pin for sensor output on 5 Hi side
AIN8	Input	Input pin for sensor output on 0,1,2 Lo side
AIN9	Input	Input pin for sensor output on 3,4,5 Lo side
REF0P	Input	DSAD+ Measurement reference voltage
REF0N	Input	DSAD- Measurement reference voltage

5.2.1 AFE・DSAD0・DSAD1

Based on the measurement conditions on Table 4-2, Table 5-4 shows the setting of DSAD0, and DSAD1, and Table 5-5 shows the setting of AFE.

Channel 0~2 of DSAD0 and DSAD1 are assigned for measurement, and channel 3~5 are assigned for Zero-reset processing.

Table 5-4 Setting of DSAD

<table>
<tr><th colspan="3" rowspan="2">Item</th><th colspan="6">Setting</th></tr>
<tr><th colspan="3">Measurement</th><th colspan="3">Zero-Reset</th></tr>
<tr><td colspan="3">ΔΣA/D Converter operation voltage setting</td><td colspan="6">3.6 V to 5.5 V (High precision)</td></tr>
<tr><td colspan="3">ΔΣA/D Converter operation mode setting</td><td colspan="6">Normal mode</td></tr>
<tr><td colspan="3">Operation clock</td><td colspan="6">PCLKB/8(4MHz)</td></tr>
<tr><td colspan="3">Start trigger source</td><td colspan="6">Software trigger</td></tr>
<tr><td colspan="3">Interrupt setting</td><td colspan="6">Not used</td></tr>
<tr><td colspan="3">Inter-unit synchronized start setting</td><td colspan="6">Enable synchronized start</td></tr>
<tr><td colspan="3">Voltage fault and disconnection detection setting</td><td colspan="6">Not used</td></tr>
<tr><td colspan="3">Channel setting</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr>
<tr><td rowspan="7">Analog input setting</td><td rowspan="2">DSAD0</td><td>+side input signal</td><td>AIN0</td><td>AIN1</td><td>AIN2</td><td>AIN0</td><td>AIN1</td><td>AIN2</td></tr>
<tr><td>- side input signal</td><td colspan="6">AIN8</td></tr>
<tr><td rowspan="2">DSAD1</td><td>+side input signal</td><td>AIN3</td><td>AIN6</td><td>AIN7</td><td>AIN3</td><td>AIN6</td><td>AIN7</td></tr>
<tr><td>- side input signal</td><td colspan="6">AIN9</td></tr>
<tr><td colspan="2">Reference input</td><td colspan="6">REF0P/REF0N</td></tr>
<tr><td colspan="2">Positive reference voltage buffer</td><td colspan="6">Disabled</td></tr>
<tr><td colspan="2">Negative reference voltage buffer</td><td colspan="6">Disabled</td></tr>
<tr><td rowspan="2">Amplifier setting</td><td colspan="2">Amplifier selection</td><td colspan="6">PGA</td></tr>
<tr><td colspan="2">PGA gain setting</td><td colspan="6">x128</td></tr>
<tr><td rowspan="9">ΔΣA/D conversion setting</td><td colspan="2">A/D conversion mode</td><td colspan="6">Normal operation</td></tr>
<tr><td colspan="2">Data format</td><td colspan="6">Two’s complement</td></tr>
<tr><td colspan="2">A/D conversion number</td><td colspan="3">Exponential operation mode, 1 time</td><td colspan="3">Immediate value mode, 64 time</td></tr>
<tr><td colspan="2">Oversampling ratio</td><td colspan="6">32</td></tr>
<tr><td colspan="2">Offset correction value</td><td colspan="3">Not set</td><td colspan="3">0</td></tr>
<tr><td colspan="2">Gain correction value</td><td colspan="6">Not set</td></tr>
<tr><td colspan="2">Enable averaging data</td><td colspan="3">Disabled</td><td colspan="3">Enabled</td></tr>
<tr><td rowspan="2"></td><td>A/D conversion end interrupt timing</td><td colspan="3" rowspan="2">-</td><td colspan="3">When the average of the selected number of value is calculated</td></tr>
<tr><td>Average data number</td><td colspan="3">64</td></tr>
<tr><td colspan="3">Disconnect detection assist setting</td><td colspan="6">Not permitted</td></tr>
</table>

Table 5-5 Setting of AFE

<table>
<tr><th colspan="2">Item</th><th>Setting</th></tr>
<tr><td rowspan="2">Bias output setting</td><td>Enable bias voltage setting</td><td>Enabled</td></tr>
<tr><td>pin output</td><td>AIN8, AIN9</td></tr>
<tr><td colspan="2">Excitation current output setting</td><td>Not used</td></tr>
<tr><td colspan="2">Low level voltage detection setting</td><td>Not used</td></tr>
<tr><td colspan="2">Low-side switch control setting</td><td>Not used</td></tr>
</table>

5.2.2 SCI1・DMAC0・DMAC3・CMT0

For communication with the PC tool program, SCI1 is used in the asynchronous mode. DMAC0 is used to obtain the received data and DMAC3 is used to set the transmitted data. In addition, CMT0 is used to detect the communication time-out.

The settings for the peripheral function are shown in the tables below.

Table 5-6 Setting of SCI1

Item		Setting
Start bit edge detection setting		Low level of RXD1 pin
Data・bit length		8bit
Parity setting		None
Stop bit length setting		1 bit
Transfer direction setting		LSB-first
Transfer rate setting	Transfer clock	Internal clock
	Bit rate	3Mbps
	Enable modulation duty correction	Enable
	SCK1 pin function	SCK1 is not used
Noise filter setting		Disable
Hardware flow control setting		CTS1#
Data handling setting	Transmit data handling	Data handling by DMAC
	Receive data handling	Data handling by DMAC
Interrupt setting	Enable reception error interrupt	Not used
Callback function Setting		Not used

Table 5-7 Setting of DMAC

Item		Setting	
		DMAC0	DMAC3
Transfer setting	Activation source	SCI1 (RXI1)	SCI1 (TXI1)
	Activation source flag control	Clear interrupt flag of the activation source	
	Transfer mode	Free running mode	Normal mode
	Transfer data size	8bit	
	Transfer count / Repeat size / Block size	-	(Setting with software)
Source address setting	Source address	0008 A025h (SCI1.RDR) Fixed	(Setting with software) Incremented
	Specify the transfer source as extended repeat area	-	Enable
	Extended repeat area		Lower 12 bits of the address (4Kbyte)
Destination address setting	Destination address	(Setting with software) Incremented	0008 A023h (SCI1.TDR) Address fixed
	Specify the transfer destination as extended repeat area	Enable	-
	Extended repeat area	Lower 9 bits of the address (512byte)	
Interrupt setting		Not permitted	

Table 5-8 Setting of CMT0

Item		Setting
Count clock setting		PCLKB/512
Compare match setting	Interval value	1000ms
	Compare match interrupt (CMI0)	Permitted Priority: Level 0 (disabled)

5.2.3 PORT

Read SW1 state using P27.

Turn on/off LED1 with PH2. Turn it on while transmitting the measurement results to the PC tool program or Zero-reset processing.

Table 5-9 shows the setting condition of PORT.

Table 5-9 Setting of PORT

Item		Setting
PORT2	P27	Input
PORTH	PH2	Output CMOS output Output 1

5.3 Zero-Reset Processing

Zero-reset processing starts when the operating mode is IDLE and SW1 is pressed. Sets the averages of 64 samples of A/D conversion values, which are obtained under the setting of DSAD for Zero-reset, to the offset correction register OFCRm of the corresponding channel to correct the offset. During the Zero-reset processing, LED1 on the RSSK board is turned on. For the settings of DSAD, refer to “Table 5-4 Setting of DSAD”.

Table 5-10 shows the operation and its process. For the processing location, refer to “Figure 5-1 Force Sensor Measurement Process Flow”.

Table 5-10 Zero-Reset Procedure

Procedure	Operation	Processing location	Process
1	Press SW1	SW1 control	・Turn on LED1 ・Set DSAD for Zero-reset and start A/D conversion ・Change the operating mode to ZERO-RESET
2	-	Zero-reset processing	・Stop A/D conversion ・Set the obtained the average of A/D conversion values in register OFCRm of the corresponding channel. ・Turn off LED1 ・Change the operating mode to IDLE

5.4 Communication Control

Based on the communication specifications of RSSKRX23R-A, process with the PC tool program are performed. A flow of communication processes is shown in Figure 5-2

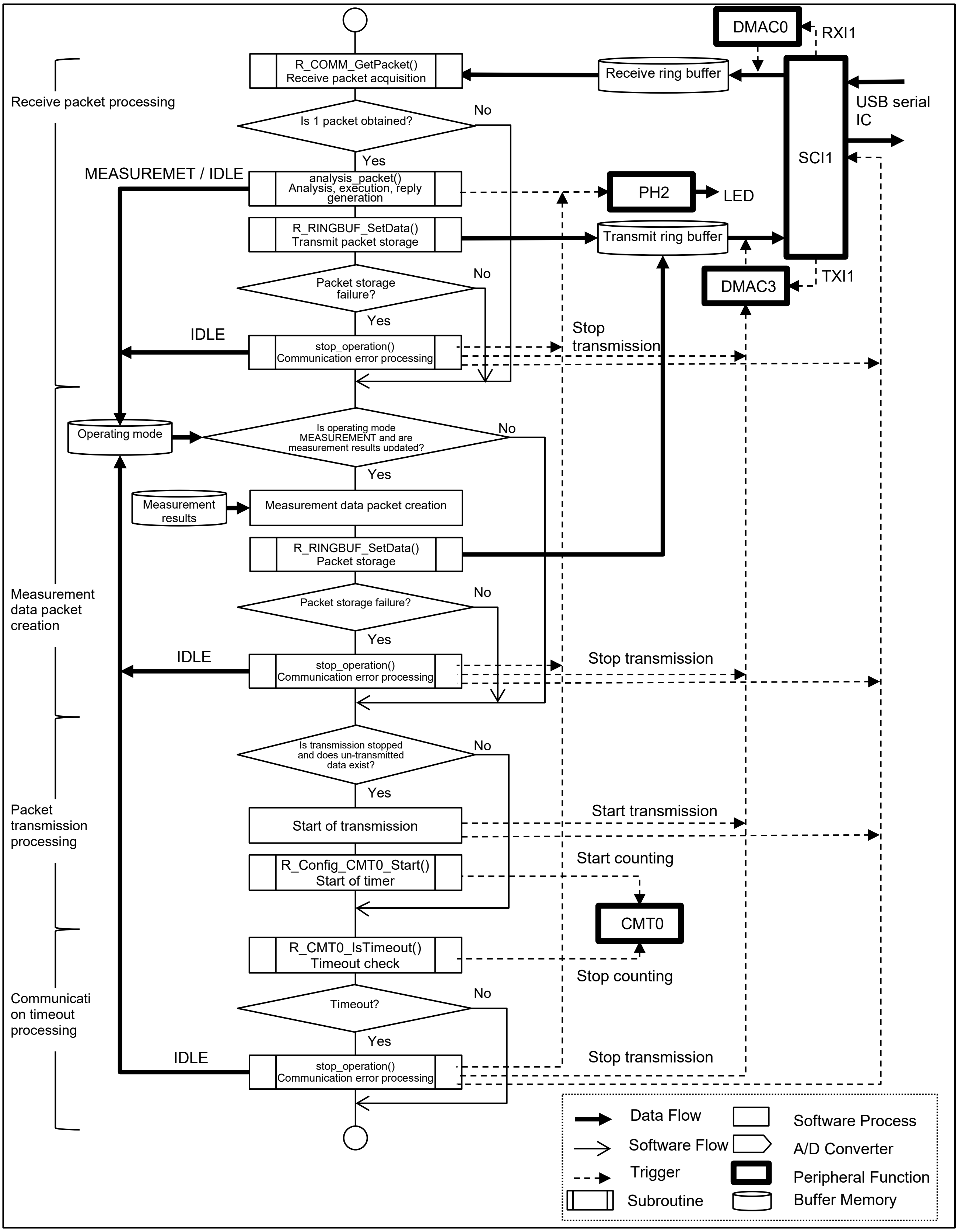


Figure 5-2 Communication Process Flow

The following provides an overview of each process.

- Receive packet processing
  Obtains a received packet from the receive ring buffer, and performs processing corresponding to a command in the packet, then creates and stores a reply packet in the transmit ring buffer. Table 5-11 lists the commands supported by this program and the processes corresponding to the commands. For an unsupported command, a NACK is returned.
  If the reply packet cannot be stored in the transmit ring buffer, communication error processing is performed.

Table 5-11 Packets and Actions

Command	Process
Negotiation	Return the software status with a reply packet
Read	Return the read value of the specified register with a reply packet
Run	・Turn on LED1 ・Set DSAD for force sensor measurement and start A/D conversion ・Change the operating mode to MEASUREMENT
Stop	・Stop A/D conversion ・Turn off LED1 ・Change the operating mode to IDLE
ExtraInformation	Return the information specified by a reply packet

- Measurement data packet creation
  If the Operating Mode is MEASUREMENT and the measurement results are updated, a TransmissionCh0 reply packet is created from the measurement results and is stored in the transmit ring buffer.
  If the reply packet cannot be stored in the transmit ring buffer, communication error processing is performed.
- Packet transmission processing
  If data is not being transmitted and the transmit ring buffer contains un-transmitted data, transmission starts with DMAC3, and 1-second counting starts with CMT0 for timeout detection.
- Communication timeout processing
  If transmission is completed, CMT0 for timeout detection is stopped.
  If transmission is in progress, the timer is checked for a compare match, and if a compare match has occurred, this is judged as a timeout. If it is judged as a timeout, communication error processing is performed.
- Communication error processing
  Whether the transmit packet cannot be stored in the transmit ring buffer or a communication timeout occurs, communication is stopped, and the following processes are performed to make a reconnection possible.
  — Stop SCI1 and DMAC3, which are used for transmission
  — Clear the transmit buffer and the measurement result transmission enable flag
  — Set Operating mode to IDLE
  — Turn LED1 OFF

Each ring buffer used for transmission and reception is for DMAC transmission, therefore, their address is arranged in the alignment adjusted for each buffer size. In this program, section name is declared “B_DMAC_REPEAT_AREA_1”, and arrangement is set based on the largest buffer size.

5.5 Program Configuration

5.5.1 File Configuration

Table 5-12 File Configuration

Folder name, File name	Description
├src	
│ ├smc_gen	Smart Configurator generation
│ │ ├general	
│ │ ├r_bsp	
│ │ ├Config_AFE	
│ │ ├Config_CMT0	
│ │ ├Config_DMAC0	
│ │ ├Config_DMAC3	
│ │ ├Config_DSAD0	
│ │ ├Config_DSAD1	
│ │ ├Config_PORT	
│ │ ├Config_SCI1	
│ │ ├r_config	
│ │ └r_pincfg	
│ ├r_ring_buffer_control_api.c	Ring buffer control program
│ ├r_ring_buffer_control_api.h	Ring buffer control API definition
│ ├r_communication_control_api.c	Communication control program
│ ├r_communication_control_api.h	Communication control API definition
│ ├r_fs_api.c	Force sensor measurement calculation program
│ ├r_fs_api.h	Force sensor measurement calculation API definition
│ ├r_fs_cfg.h	Force sensor measurement condition definition
│ └main.c	Main processing
└dsplib-rxv2	RX DSP library file

5.5.2 Macro Definition

Table 5-13 main.c Definitions

Definition Name	Type	Initial value	Description
D_PC_TOOL_USE	bool	1	Communication with PC tool program 0: Not used 1: Used

Table 5-14 r_fs_cfg.h: Definitions for Force Sensor Measurement

Definition Name	Type	Initial value	Description
D_FS_CFG_GAIN	float	8.0F	PGA gain G_{PGA} [time]
D_FS_CFG_VREF	float	5.0F	Reference Voltage of A/D conversion V_{REF} [V]
D_FS_CFG_DSADRES	int	24	Resolution of A/D conversion [bit]
D_FS_CFG_CHANNELS	int	6	Number of input channels

5.5.3 Structure

Table 5-15 r_ring_buffer_control_api.h: Structure for Ring Buffer Control

Structure type	st_ring_buf_t		
Member Variable	**Type**	**Name**	**Description**
	uint8_t *	buf	Ring buffer pointer
	size_t	length	Ring buffer length
	uint32_t	r_index	Read index
	uint32_t	w_index	Write index

5.5.4 Functions

Table 5-16 main.c Fuctions

Name	**main**			
Description	main function			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	O	void	-	
Name	**analysis_packet**			
Description	Execute command corresponding to receive packet, and store response packet. In the case of Run/Stop command, update the Operating Mode.			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	uint8_t const	rcv_pck[]	Receive packet storage array
	O	uint8_t	send_pkt[]	Reply packet storage array
	I/O	e_mode_t *	p_mode	Pointer to the Operating mode variable
Return Value	O	size_t	Response packet length [byte]	
Name	**stop_operation**			
Description	Stop DMAC and SCI, initialize ring buffer, Turn off LED1			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I/O	st_ring_buf_t *	ary	Pointer to ring buffer
Return Value	-	void	-	

Table 5-17 r_fs_api Functions

Name	**R_FS_DsadToVoltage**			
Description	Convert DSAD value to input voltage			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	float	dsad	24bit DSAD value
Return Value	O	float	Voltage [V]	

Table 5-18 r_communication_control_api Functions

Name	**R_COMM_GetPacket**			
Description	Read 1 packet from receive ring buffer			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	st_ring_buf_t *	r_buf	Pointer to receive ring buffer
	O	uint8_t	r_packet[]	Receive packet storage array
Return Value	O	size_t	Packet length [byte]	

Table 5-19 r_ring_buffer_control_api Functions

Name	**R_RINGBUF_GetData**			
Description	Read the specified byte number from ring buffer			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	st_ring_buf_t *	ary	Pointer to ring buffer
	O	uint8_t	data[]	Data storage array
	I	size_t	len	Number of bytes to read
	I	bool	index_update	Index update flag true: updated false: not updated
Return Value	O	size_t	Number of read bytes	
Name	**R_RINGBUF_SetData**			
Description	Write the specified byte number to ring buffer			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I/O	st_ring_buf_t *	ary	Pointer to ring buffer
	I	uint8_t	data[]	Data storage array
	I	size_t	len	Number of bytes to write
Return Value	O	size_t	Number of written bytes	
Name	**R_RINGBUF_GetDataLength**			
Description	Read the number of bytes stored in ring buffer			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	st_ring_buf_t *	ary	Pointer to ring buffer
Return Value	O	size_t	Number of stored bytes	
Name	**R_RINGBUF_SetDataIndex**			
Description	Update the index of ring buffer			
Arguments	**I/O**	**Type**	**Name**	**Description**
	O	st_ring_buf_t *	ary	Pointer to ring buffer
	I	uint16_t	value	Index value
	I	uint8_t	select	Target index 0: Read index 1: Write index
Return Value	O	uint32_t	Index value	

Table 5-20 Config_CMT0 User Defined Functions

Name	**R_CMT0_IsTimeout**			
Description	Return if time-out or not			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	bool	flag	Counter stare false: continued true: stopped
Return Value	O	bool	false: Counting true: Time-out	
Name	**R_CMT0_CntClear**			
Description	Clear the compare match timer counter			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	-	void	-	

Table 5-21 Config_DMAC0 User Defined Functions

Name	**R_DMAC0_SetDestAddr**			
Description	Set destination address to DMDAR			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	void *	p_addr	Destination address
Return Value	-	void	-	
Name	**R_DMAC0_GetDestAddr**			
Description	Get destination address (macro function)			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	O	void *	DMDAR value	

Table 5-22 Config_DMAC3 User Defined Functions

Name	**R_DMAC3_SetSrcAddr**			
Description	Set source address to DMSAR			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	void *	p_addr	Source address
Return Value	-	void	-	
Name	**R_DMAC3_SetTxCnt**			
Description	Set transfer count to DMCRA			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	uint32_t	cnt	Transfer count
Return Value	-	void	-	

Table 5-23 Config_DSAD0 User Defined Functions

Name	**R_DSAD0_IsConversionEnd**			
Description	Returns whether A/D conversion is in progress.			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	-	bool	false: Converting true: Conversion end	
Name	**R_DSAD0_ClearConversionEndFlag**			
Description	Clear ADI0 flag			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	-	void	-	
Name	**R_DSAD0_IsScanEnd**			
Description	Returns whether auto scan is in progress.			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	O	bool	false: Scanning true: Scan end	
Name	**R_DSAD0_ClearScanEndFlag**			
Description	Clear SCANEND0 flag			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	-	void	-	
Name	**R_DSAD0_GetADValue**			
Description	Return DR register value (macro function)			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	O	uint32_t	DR value	
Name	**R_DSAD0_GetAverageADValue**			
Description	Return AVDR register value (macro function)			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	O	uint32_t	AVDR value	

Table 5-24 Config_DSAD0 User Defined Functions (continue)

Name	**R_DSAD0_SetOFCR0**			
Description	Set offset correction value to OFCR0 (macro function)			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	uint32_t	val	Setting value to OFCR0
Return Value	-	void	-	
Name	**R_DSAD0_SetOFCR1**			
Description	Set offset correction value to OFCR1 (macro function)			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	uint32_t	val	Setting value to OFCR1
Return Value	-	void	-	
Name	**R_DSAD0_SetOFCR2**			
Description	Set offset correction value to OFCR1 (macro function)			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	uint32_t	val	Setting value to OFCR2
Return Value	-	void	-	
Name	**R_Config_DSAD0_CHnEN**			
Description	Set A/D Conversion Enable bit to MR			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	uint32_t	ch	Permission setting of channel 0-5 to bit 0-5 1: Conversion enable 0: Conversion disable
Return Value	-	void	-	

Table 5-25 Config_DSAD1 User Defined Functions

Name	**R_DSAD1_IsConversionEnd**			
Description	Returns whether A/D conversion is in progress.			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	-	bool	false: Converting true: Conversion end	
Name	**R_DSAD1_ClearConversionEndFlag**			
Description	Clear ADI1 flag			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	-	void	-	
Name	**R_DSAD1_IsScanEnd**			
Description	Returns whether auto scan is in progress.			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	O	bool	false: Scanning true: Scan end	
Name	**R_DSAD1_ClearScanEndFlag**			
Description	Clear SCANEND1 flag			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	-	void	-	

Table 5-26 Config_DSAD1 User Defined Functions (continue)

Name	**R_DSAD1_GetADValue**			
Description	Return DR register value (macro function)			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	O	uint32_t	DR value	
Name	**R_DSAD1_GetAverageADValue**			
Description	Return AVDR register value (macro function)			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	O	uint32_t	AVDR value	
Name	**R_DSAD1_SetOFCR0**			
Description	Set offset correction value to OFCR0 (macro function)			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	uint32_t	val	Setting value to OFCR0
Return Value	-	void	-	
Name	**R_DSAD1_SetOFCR1**			
Description	Set offset correction value to OFCR1 (macro function)			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	uint32_t	val	Setting value to OFCR1
Return Value	-	void	-	
Name	**R_DSAD1_SetOFCR2**			
Description	Set offset correction value to OFCR2 (macro function)			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	uint32_t	val	Setting value to OFCR2
Return Value	-	void	-	
Name	**R_Config_DSAD1_ChnEN**			
Description	Set A/D Conversion Enable bit to MR			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	uint32_t	ch	Permission setting of channel 0-5 to bit 0-5 1: Conversion enable 0: Conversion disable
Return Value	-	void	-	

Table 5-27 Config_PORT User Defined Functions

Name	**R_LED1_On**			
Description	Turn on LED1 (Macro function)			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	-	void	-	
Name	**R_LED1_Off**			
Description	Turn off LED1 (Macro function)			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	-	void	-	
Name	**R_PORT_KeyScan**			
Description	Acquires the status of switch SW1 that has absorbed chattering			
Arguments	**I/O**	**Type**	**Name**	**Description**
	I	uint32_t	key_current	Previous SW1 status
Return Value	O	uint32_t	SW1 status 0: On 1: Off	

Table 5-28 Config_SCI1 User Defined Functions

Name	**R_SCI1_IsTransferEnd**			
Description	Returns the transfer status			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	O	bool	false: Transferring true: Transfer end	
Name	**R_SCI1_SendStart**			
Description	Start transmission			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	O	MD_STATUS	MD_OK	
Name	**R_SCI1_SendStop**			
Description	Stop transmission			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	O	MD_STATUS	MD_OK	
Name	**R_SCI1_ReceiveStart**			
Description	Start receiving			
Arguments	**I/O**	**Type**	**Name**	**Description**
	-	void	-	-
Return Value	O	MD_STATUS	MD_OK	

6. Importing a Project

After importing the sample project, make sure to confirm build and debugger setting.

6.1 Importing a Project into e2 studio

Follow the steps below to import your project into e^2 studio. Pictures may be different depending on the version of e^2 studio to be used.

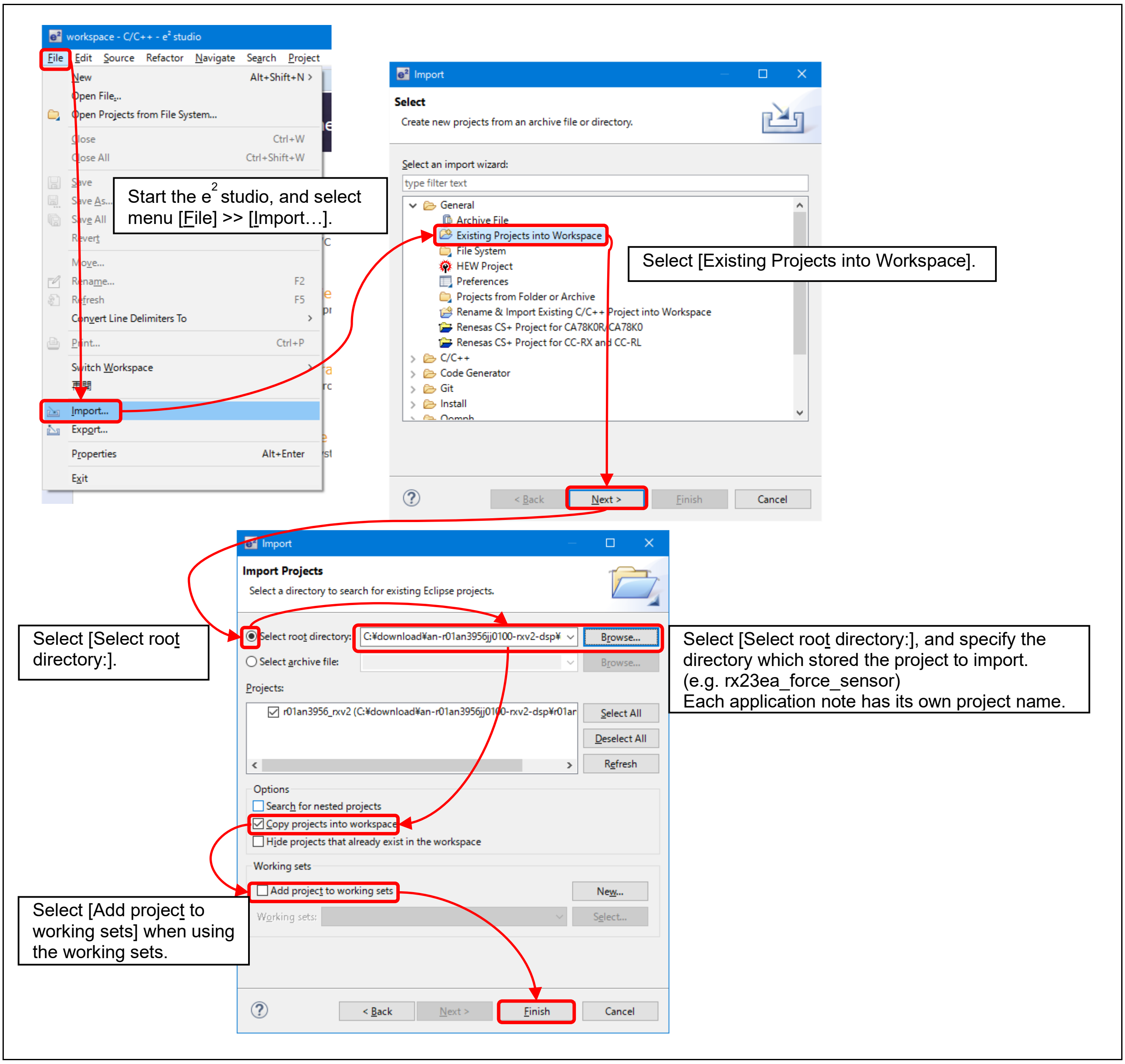


Figure 6-1 Importing a Project into e^2 studio

6.2 Importing a Project into CS+

Follow the steps below to import your project into CS+. Pictures may be different depending on the version of CS+ to be used.

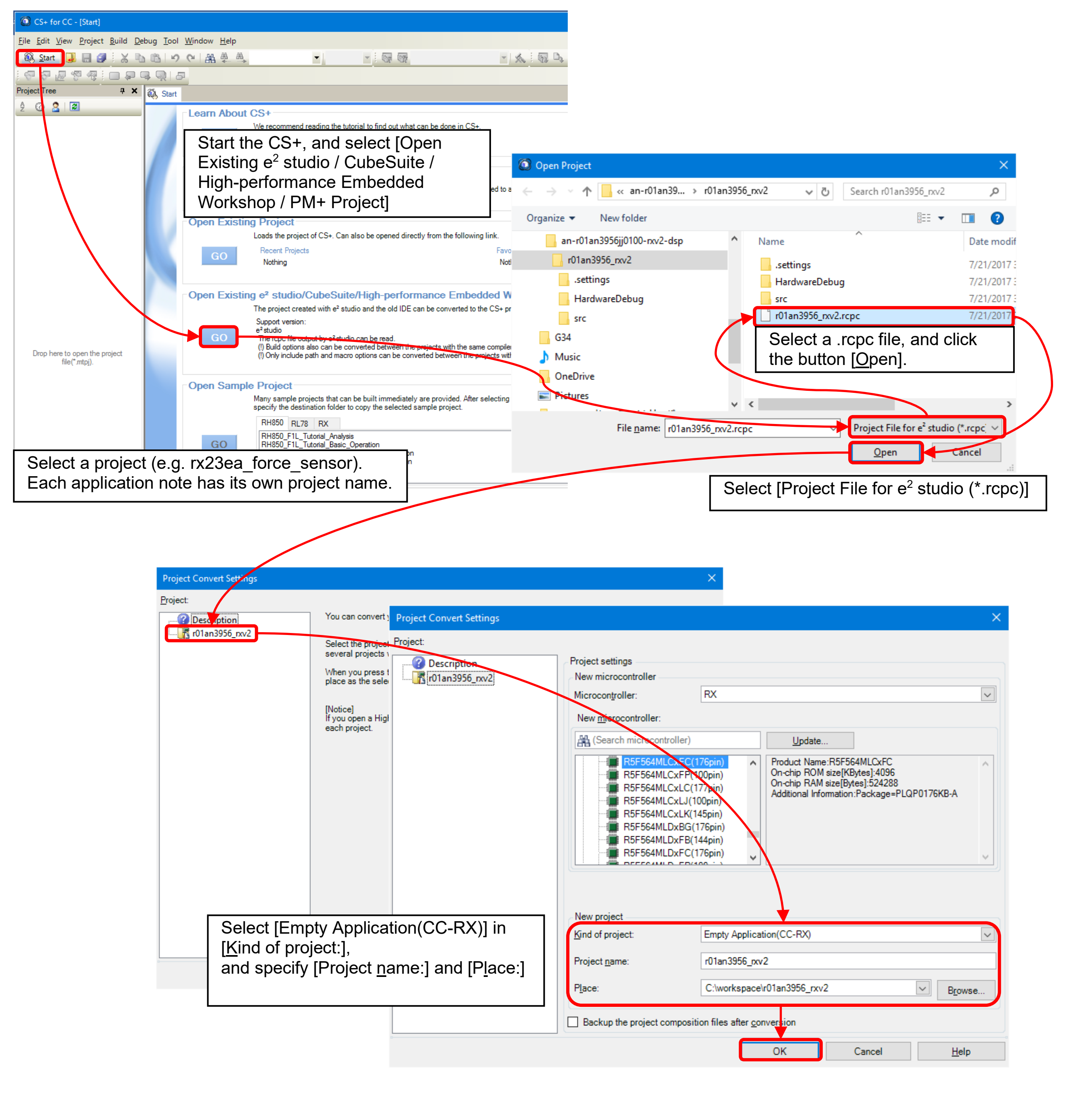


Figure 6-2 Importing a Project into CS+

7. Measurement Result using Sample Program

7.1 Memory Usage and Execution Cycle

7.1.1 Build Conditions

, Table 7-1 shows the build conditions of sample program under environment shown in “3.Environment for Operation Confirmation”. This setting is default setting when project is generated, except for memory allocation to support the PC tool.

Table 7-1 Build Conditions

Item		Setting
Compiler	Not supporting PC tool	-isa=rxv2 -include=".¥dsplib-rxv2"-utf8 -nomessage -debug -outcode=utf8 -nologo
	Supporting PC tool	add to the above -define=D_PC_TOOL_USE=1
Linker		-library=".¥dsplib-rxv2¥RX_DSP_FPU_LE.lib" -noprelink -output="rx23ea_force_sensor.abs" -form=absolute -nomessage -vect=_undefined_interrupt_source_isr -list=rx23ea_force_sensor.map -nooptimize -rom=D=R,D_1=R_1,D_2=R_2 -nologo
	Added section	-start=B_DMAC_REPEAT_AREA_1/02000

Note: Include paths other than user settings in compiler setting are omitted.

7.1.2 Memory Usage

The amount of memory usage of sample program is shown in Table 7-2.

Table 7-2 Amount of Memory Usage

Item		Size [byte]		Remarks
		Not supporting PC tool	Supporting PC tool	
ROM		9980	10558	
	Code	8029	8559	
	Data	1951	1999	
RAM		7114(2118)	12284(7288)	Note
	Data	1994	7164	
	Stack	5120(124)	5120(124)	Note

Note: RAM usage for stack is shown in “()”

7.1.3 The number of Execution Cycle

The number of execution cycles and processing load for each block in “Figure 5-1 Force Sensor Measurement Process Flow” is shown in Table 7-3.

Table 7-3 Number of Execution Cycle

ICLK=32MHz

Item	Number of execution cycle (Execution time)	Process load [%]	Condition
Measurement・Calculation	568cycle (17.75µsec)	1.16	Maximum cycles at operating mode MEASUREMENT
Communication control	710cycle (22.19µsec)	1.44	Maximum cycles at normal operation
SW1 control	307cycle (9.59µsec)	0.62	Maximum cycles at normal operation

Note: Process load is calculated based on the execution time of DSAD output cycle (1.536msec).

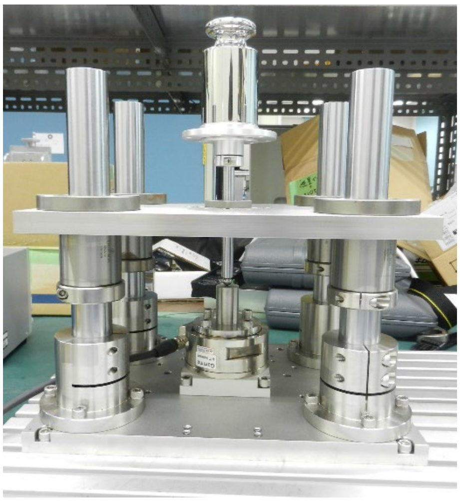
7.2 Force Sensor Measurement

7.2.1 Measurement Appearance

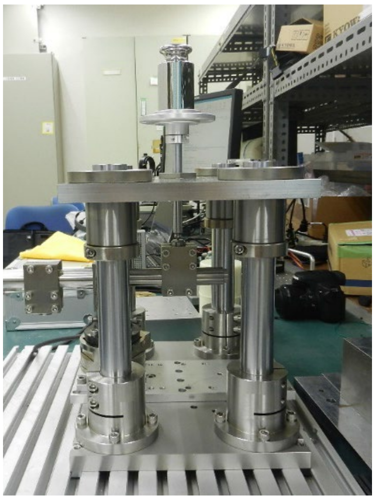
Connecting a force sensor based on the configuration in “Figure 4-1 Connection of RSSKRX23E-A Board and Force Sensor”, we have performed measurement applying force and torque to the force sensor with evaluation jigs and weights. Figure 7-1 shows the appearance of this measurement.

Applying to F_x, F_y

Applying to F_z



Applying to T_x, T_y



Applying to T_z

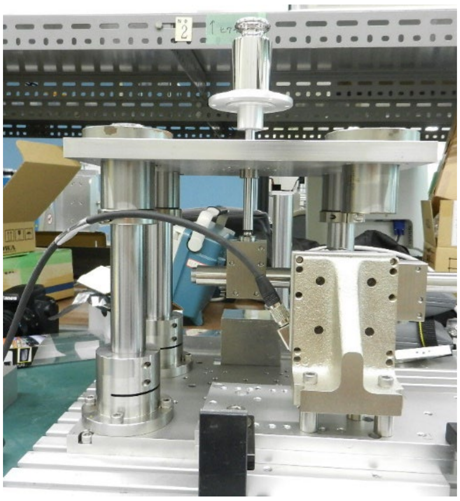


Figure 7-1 Measurement Appearance

7.2.2 Measurement Condition

Figure 7-2 and Figure 7-3 show how to apply force and torque, and Figure 7-4 shows the weights used in measurement.

For measurement, Zero-reset is processed in the posture shown in Figure 7-2 and Figure 7-3 at no load.

(1) Force Measurement

Force F [N] applied to a force sensor is calculated from weight m [kg] and gravitational acceleration g [m/s^2] with the equation below.

$$F = m \times g$$

(2) Torque Measurement

Torque T [N·m] applied to a force sensor is calculated from weight m [kg], gravitational acceleration g [m/s^2], and the distance between a fulcrum and a force point L [m] with the equation below.

$$T = m \times g \times L$$

Suppose that gravitational acceleration is the standard gravitational acceleration 9.80665[m/s2].

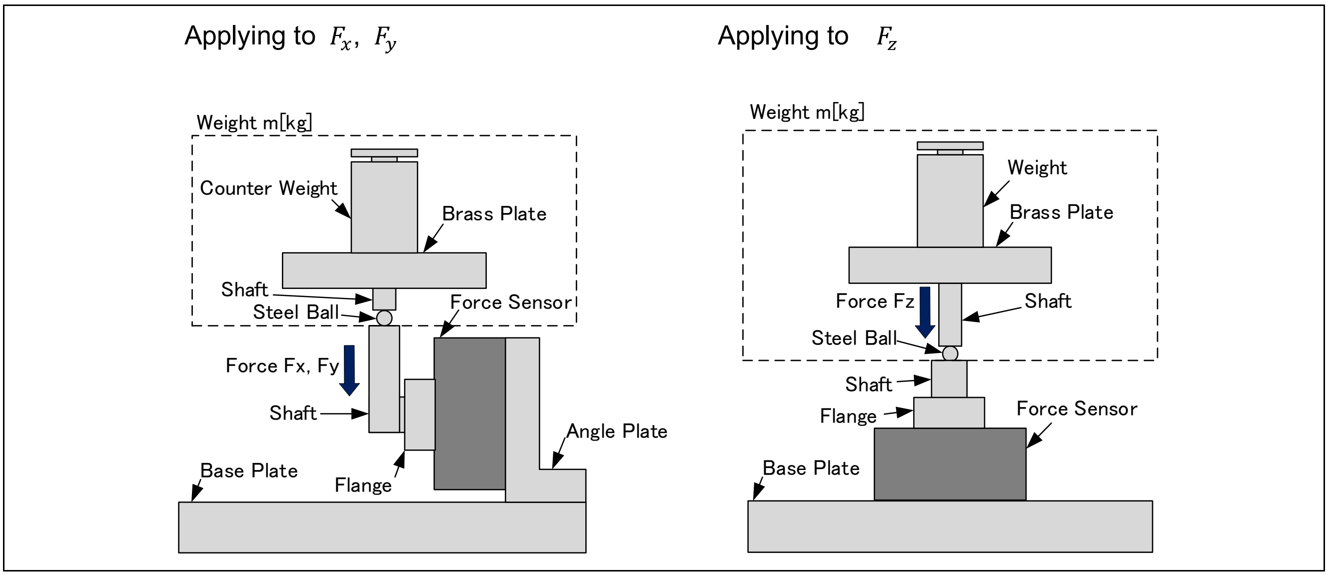


Figure 7-2 How to Apply Force

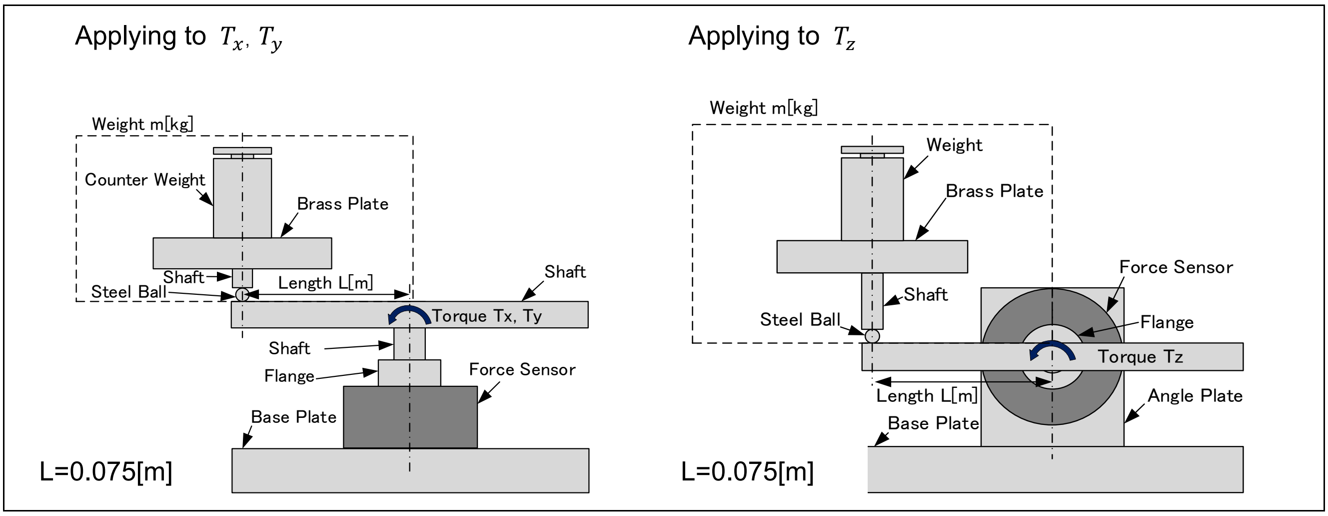


Figure 7-3 How to Apply Torque

Table 7-4 Weight Used in Measurement

No.	Name	Model	Weight	Grade	Manufacturer
1	Weight Set	WS1M1K	1mg x1, 2mg x2, 5mg x1 10mg x1, 20mg x2, 50mg x1 100mg x1, 200mg x2, 500mg x1 1g x1, 2g x2, 5g x1 10g x1, 20g x2, 50g x1 100g x1, 200g x2, 500g x1 1kg x1	M1	AS ONE
2	Cylindrical Weight	SWM2000	2kg	M1	AS ONE
3	Brass Plate	INERTIAPLATE: C	2.853kg[Note]	-	Renesas
4	Brass Plate	INERTIAPLATE: D	4.6625kg[Note]	-	Renesas

Note: Confirmed with A&D counting scale FC-5000i (Repeatability 0.5g)

7.2.3 Measurement Result

The result of force measurement is shown in Figure 7-4, and the result of torque measurement is shown in Figure 7-5. The measurement results are corrected by calculating scale factor error and bias error from the measurement values at no load and at maximum load.

From the measurement result, the force measurement error $E_{F:FS}$ for full-scale is calculated from the force input value F_{in}, the force measurement value F_{mea}, and the force measurement range of the force sensor F_{FS} (F_x, F_y :130N, F_z :400N) with the equation below.

$$E_{F:FS} = \frac{F_{mea} - F_{in}}{F_{FS}} \times 100[\%FS]$$

Similarly, the torque measurement error $E_{T:FS}$ is calculated from the torque input value T_{in}, the torque measurement value T_{mea}, the torque measurement range of the force sensor T_{FS} (T_x, T_y, T_z:10N·m) with the equation below.

$$E_{T:FS} = \frac{T_{mea} - T_{in}}{T_{FS}} \times 100[\%FS]$$

Table 7-5 shows the measurement uncertainty of the force sensor 9105-TWE-Gamma used in this measurement and the full-scale error of this measurement. These errors are indicators showing the linearity of the measurement.

Table 7-5 shows that the force measurement error is within ±0.25% FS, and the torque measurement error is within ±1% FS, indicating that these errors are within the measurement uncertainty of the force sensor used in this measurement. Though this result contains not only the error of the circuit and the nonlinearity of the force sensor itself, but also flexure or inclination of the evaluation jigs and the error caused by friction, it is confirmed that this system configuration allows the measurement of the force sensor.

Table 7-5 Measurement Uncertainty

Item	$E_{Fx:FS}$ [%FS]	$E_{Fy:FS}$ [%FS]	$E_{Fz:FS}$ [%FS]	$E_{Tx:FS}$ [%FS]	$E_{Ty:FS}$ [%FS]	$E_{Tz:FS}$ [%FS]
9105-TWE-Gamma SI-130-10 Measurement uncertainty (95% CI)	1.00%	1.25%	0.75%	1.00%	1.25%	1.50%
Result of full-scale error measurement (Worst case)	0.13%	0.14%	0.07%	0.95%	0.68%	0.89%

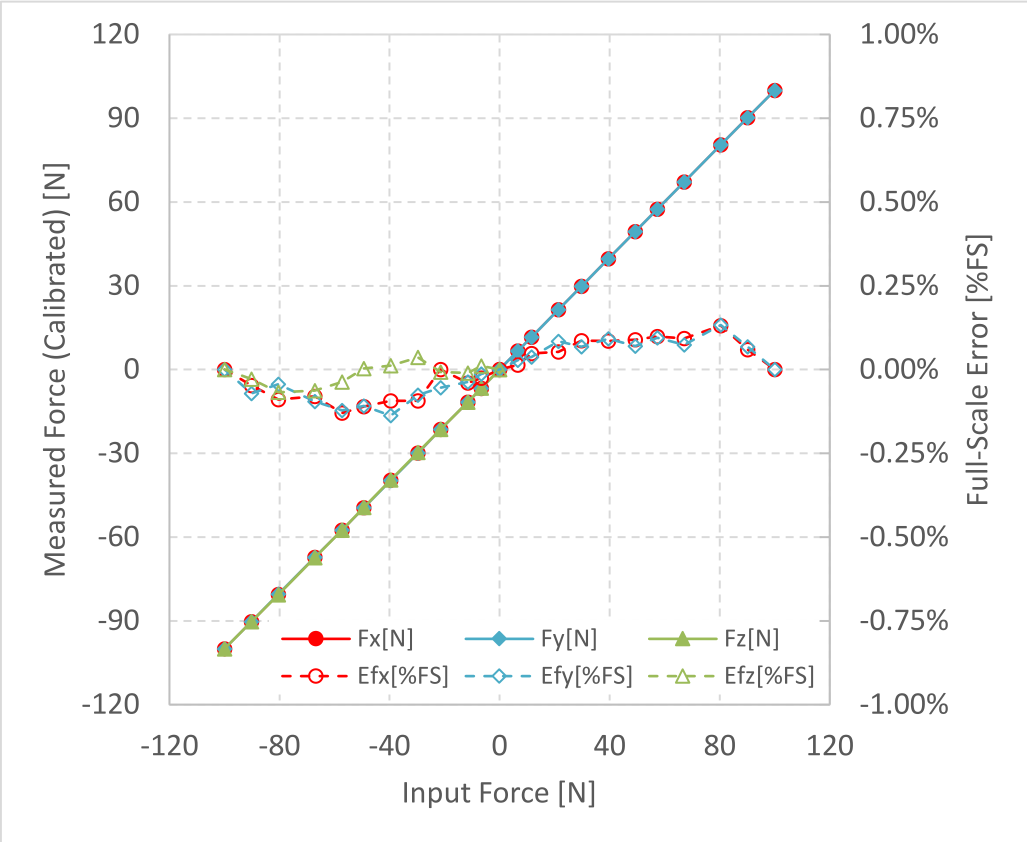


Figure 7-4 Force Measurement Result

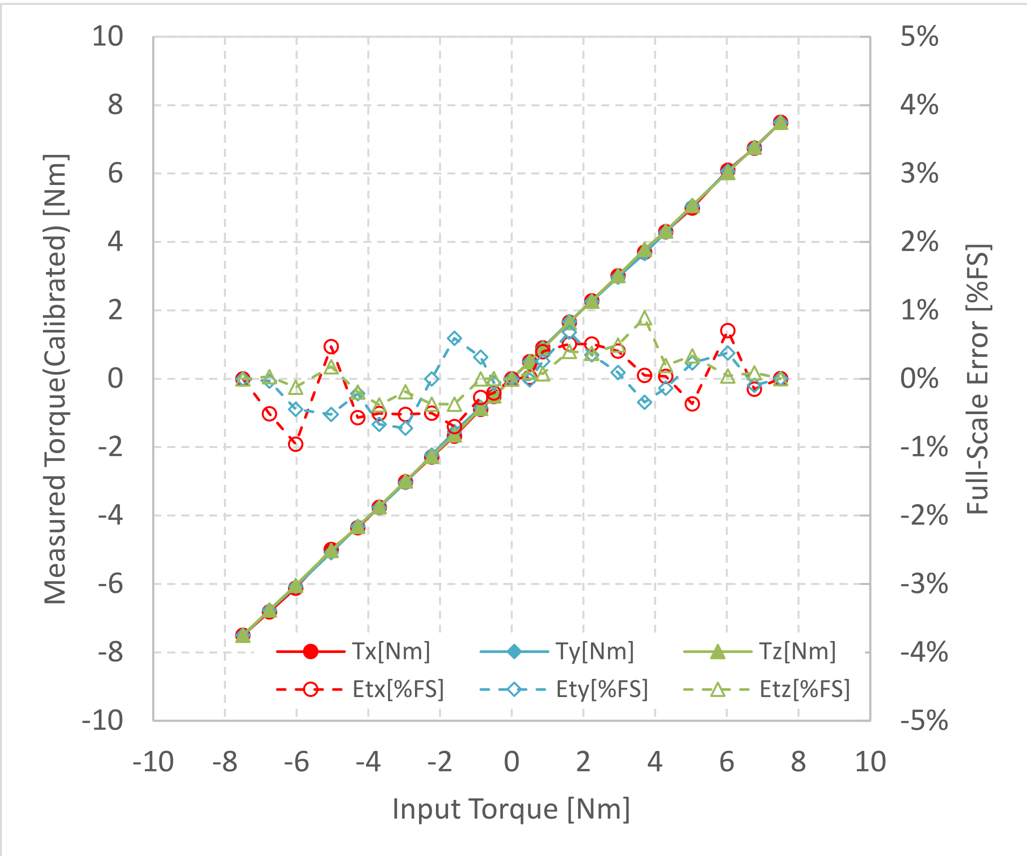


Figure 7-5 Torque Measurement Result

Revision History

Rev.	Date	Description	
		Page	Summary
Rev.1.00	Nov.15.21	-	-

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.
5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.
6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit: www.renesas.com/contact/.